



UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/823,845	04/14/2004	David Hsing Lin	200402290-1	5529
22879	7590	10/28/2010	EXAMINER	
HEWLETT-PACKARD COMPANY Intellectual Property Administration 3404 E. Harmony Road Mail Stop 35 FORT COLLINS, CO 80528				AHLUWALIA, NAVNEET K
2166		ART UNIT		PAPER NUMBER
			NOTIFICATION DATE	
			DELIVERY MODE	
			10/28/2010	
			ELECTRONIC	

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Notice of the Office communication was sent electronically on above-indicated "Notification Date" to the following e-mail address(es):

JERRY.SHORMA@HP.COM
ipa.mail@hp.com
laura.m.clark@hp.com



UNITED STATES PATENT AND TRADEMARK OFFICE

Commissioner for Patents
United States Patent and Trademark Office
P.O. Box 1450
Alexandria, VA 22313-1450
www.uspto.gov

**BEFORE THE BOARD OF PATENT APPEALS
AND INTERFERENCES**

Application Number: 10/823,845

Filing Date: April 14, 2004

Appellant(s): LIN, DAVID HSING

Karen G. Hazzah
For Appellant

EXAMINER'S ANSWER

This is in response to the appeal brief filed 1/8/2010 appealing from the Office action mailed 5/14/2009.

(1) Real Party in Interest

The examiner has no comment on the statement, or lack of statement, identifying by name the real party in interest in the brief.

(2) Related Appeals and Interferences

The examiner is not aware of any related appeals, interferences, or judicial proceedings which will directly affect or be directly affected by or have a bearing on the Board's decision in the pending appeal.

(3) Status of Claims

The following is a list of claims that are rejected and pending in the application:

1 – 22.

(4) Status of Amendments After Final

The examiner has no comment on the appellant's statement of the status of amendments after final rejection contained in the brief.

(5) Summary of Claimed Subject Matter

The examiner has no comment on the summary of claimed subject matter contained in the brief.

(6) Grounds of Rejection to be Reviewed on Appeal

The examiner has no comment on the appellant's statement of the grounds of rejection to be reviewed on appeal. Every ground of rejection set forth in the Office action from which the appeal is taken (as modified by any advisory actions) is being maintained by the examiner except for the grounds of rejection (if any) listed under the

Art Unit: 2166

subheading "WITHDRAWN REJECTIONS." New grounds of rejection (if any) are provided under the subheading "NEW GROUNDS OF REJECTION."

(7) Claims Appendix

The examiner has no comment on the copy of the appealed claims contained in the Appendix to the appellant's brief.

(8) Evidence Relied Upon

US 6,898,650 Gao et al. 05-2005

US 6,449,614 Scott Marcotte 09-2002

(9) Grounds of Rejection

The following ground(s) of rejection are applicable to the appealed claims:

Claim Rejections - 35 USC § 103

1. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

2. Claims 1 – 22 are rejected under 35 U.S.C. 103(a) as being unpatentable over Gao et al. ('Gao' herein after) (US 6,898,650 B1) further in view of Scott T. Marcotte ('Marcotte' herein after) (US 6,449,614 B1).

With respect to claim 1,

Gao discloses a method for retrieving data comprising: locking a linked list (column 2 lines 46 – 54, Gao); retrieving data from an element in the linked list and also advancing to a subsequent element while a breakpoint is not encountered (figure 3 and column 3 lines 39 – 67, Gao); marking the subsequent element in the linked-list as in-use when a breakpoint is encountered (column 3 lines 39 – 50, Gao); creating a recommencement reference to the subsequent element (column 4 lines 36 – 49 and 62 – 67, Gao discloses the breakpoint being encountered using figure 5 A&B using elements 510 and 525, creating a recommencement reference to a subsequent element in Gao is clearly found in column 2 lines 46 – 58, column3 lines 9 – 16 and lines 56 – 59. when the breakpoint has been marked and the flag set as in-use the recommencement point is the one at breakpoint when the flag is unset or search for another container is made according the algorithm in figure 5A&B) and unlocking the linked list (column 4 lines 60 – 66, Gao).

Gao however does not disclose the relinquishing of control as a requirement as explicitly being claimed in definition of breakpoint.

Marcotte discloses the condition where relinquishing of control may be required and this is explained in figure 2 and column 7 lines 26 – 65. Here Marcotte discloses how the lock that is help can be in a priority state and thus the queue would have to wait, similarly if the queue that was requiring the lock was in priority state the current item would have to relinquish the lock.

It would have been obvious to one of ordinary skill in the art of data processing at the time of the present invention to combine the teachings of cited references because

both inventions are directed to the same field of study of lists and locks in data processing. Furthermore, the state of priority and the lock and its release capabilities of Marcotte reduce lock contention of Gao's method (column 4 lines 4 – 39, Marcotte).

3. Claims 2 – 4 are rejected under the same rationale as claim 1 above. For further citations and references see below.

With respect to claim 2,

Gao as modified discloses the method of claim 1 further comprising: locking the linked list (column 2 lines 46 – 54, Gao); determining a subsequent element in the linked list according to the commencement reference (column 5 lines 1 – 9, Gao); and retrieving data from the determined subsequent element (column 5 lines 10 – 17, Gao).

With respect to claim 3,

Gao as modified discloses the method of claim 1 wherein creating a commencement reference to the subsequent element comprises: retrieving a pointer to the subsequent element (column 2 lines 46 – 54, Gao); determining a process identifier for a current process (column 2 lines 64 – 67 and column 3 lines 1 – 8, Gao); and associating the pointer with the process identifier (column 3 lines 64 – 67 and column 4 lines 1 – 9, Gao).

With respect to claim 4,

Gao as modified discloses the method of claim 1 wherein marking the subsequent element in the linked-list as in-use comprises maintaining a count of the quantity of processes that require additional access to the element (figure 3 and column 3 lines 39 – 51, Gao).

With respect to claim 5,

Gao discloses a method for deleting an element from a linked list comprising: determining if the element to be deleted is in-use (column 5 lines 10 – 21, Gao); updating a commencement reference to the element to refer to a data element that is subsequent to the data element to be deleted when the element in is in-use (Table 14, Gao); and deleting the element (column 5 lines 25 – 43, Table 14, Gao).

Gao however does not disclose the relinquishing of control as a requirement as explicitly being claimed in definition of breakpoint.

Marcotte discloses the condition where relinquishing of control may be required and this is explained in figure 2 and column 7 lines 26 – 65. Here Marcotte discloses how the lock that is help can be in a priority state and thus the queue would have to wait, similarly if the queue that was requiring the lock was in priority state the current item would have to relinquish the lock.

It would have been obvious to one of ordinary skill in the art of data processing at the time of the present invention to combine the teachings of cited references because both inventions are directed to the same field of study of lists and locks in data

processing. Furthermore, the state of priority and the lock and its release capabilities of Marcotte reduce lock contention of Gao's method (column 4 lines 4 – 39, Marcotte).

4. Claim 6 is rejected under the same rationale as claim 5 above. For further citations and references see below.

With respect to claim 6,

Gao as modified discloses the method of claim 5 wherein updating a recommencement reference to the element comprises: discovering a pointer associated with a process identifier (column 5 lines 1 – 9, Gao); disassociating the process identifier from the pointer; determining a pointer to a subsequent element (column 5 lines 10 – 17, Gao); and associating the process identifier with the newly determined pointer (column 2 lines 64 – 67 and column 3 lines 1 – 8, Gao).

With respect to claim 7,

Gao discloses an apparatus for storing and retrieving data comprising: processor capable of executing an instruction sequence, memory for storing an instruction sequence, input unit for receiving data (Figures 1, 5A and 5 B, Gao); first output unit for providing data according to a received data request, one or more ancillary output units for providing data according to a received data request (column 2 lines 46 – 54, Gao); instruction sequences stored in the memory including: data storage module that, when executed by the processor, minimally causes the processor to: receive data from the input unit allocate a data element to accommodate the data create a reference to the

Art Unit: 2166

data element (column 5 lines 1 – 9, Gao); store the reference in at least one of a header pointer and a forward pointer included in a preceding data element and store the data in the data element (column 1 lines 29 – 43, Gao); data service module that, when executed by the processor, minimally causes the processor to: recognize a data request from the first output unit to the exclusion of all other data requests (column 3 lines 51 – 62, Gao); provide data to the first output unit from a data element according to a data element reference and also advance the data element reference to a subsequent data element while a breakpoint is not encountered (column 2 lines 31 – 38, Gao); mark a subsequent data element as in-use when a breakpoint is encountered (column 3 lines 39 – 50, Gao); create a commencement reference to a subsequent data element column 4 lines 36 – 49 and 62 – 67, Gao discloses the breakpoint being encountered using figure 5 A&B using elements 510 and 525, creating a commencement reference to a subsequent element in Gao is clearly found in column 2 lines 46 – 58, column3 lines 9 – 16 and lines 56 – 59. when the breakpoint has been marked and the flag set as in-use the commencement point is the one at breakpoint when the flag is unset or search for another container is made according the algorithm in figure 5A&B); and enable recognition of other data requests (column 4 lines 60 – 66, Gao).

Gao however does not disclose the relinquishing of control as a requirement as explicitly being claimed in definition of breakpoint.

Marcotte discloses the condition where relinquishing of control may be required and this is explained in figure 2 and column 7 lines 26 – 65. Here Marcotte discloses how the lock that is help can be in a priority state and thus the queue would have to

Art Unit: 2166

wait, similarly if the queue that was requiring the lock was in priority state the current item would have to relinquish the lock.

It would have been obvious to one of ordinary skill in the art of data processing at the time of the present invention to combine the teachings of cited references because both inventions are directed to the same field of study of lists and locks in data processing. Furthermore, the state of priority and the lock and its release capabilities of Marcotte reduce lock contention of Gao's method (column 4 lines 4 – 39, Marcotte).

5. Claims 8 – 12 are rejected under the same rationale as claim 7 above. For further citations and references see below.

With respect to claim 8,

Gao as modified discloses the apparatus of claim 7 wherein the data service module, when executed by the processor, further minimally causes the processor to: recognize a data request from the first output unit to the exclusion of all other data requests (column 2 lines 46 – 54, Gao); and provide data to the first output unit from a data element according to the commencement reference (column 5 lines 1 – 9, Gao).

With respect to claim 9,

Gao as modified discloses the apparatus of claim 7 wherein the data service module causes the processor to create a commencement reference by minimally causing the processor to: retrieve a pointer to a data element subsequent to a current data element (column 2 lines 64 – 67 and column 3 lines 1 – 8, Gao); determine an

identifier associated with the data request received from the first output unit and store the retrieved pointer and the determined identifier in an associative manner (column 3 lines 64 – 67 and column 4 lines 1 – 9, Gao).

With respect to claim 10,

Gao as modified discloses the apparatus of claim 7 wherein the data service module causes the processor to mark a subsequent data element as in-use by minimally causing the processor to increment a use counter included in a subsequent data element (figure 3 and column 3 lines 39 – 51, Gao).

With respect to claim 11,

Gao as modified discloses the apparatus of claim 7 wherein the data service module further minimally causes the processor to receive a delete data request from an output unit by minimally causing the processor to: determine if a data element to be deleted is in-use (column 5 lines 10 – 21, Gao); update a commencement reference to refer to a data element that is subsequent to the data element to be deleted (Table 14, Gao); and delete the data element according to the received delete data request (column 5 lines 25 – 43, Table 14, Gao).

With respect to claim 12,

Gao as modified discloses the apparatus of claim 11 wherein the data service module causes the processor to update a commencement reference by minimally

causing the processor to: discover a pointer according to a data request identifier (column 5 lines 1 – 9, Gao); and replace the pointer with a pointer to a data element that is subsequent to the data element to be deleted (column 5 lines 10 – 17, Gao).

With respect to claim 13,

Gao discloses a computer readable medium having imparted thereon one or more instruction sequences for storing and retrieving data comprising: data storage module that, when executed by a processor, minimally causes the processor to: receive data from an input unit, allocate a data element to accommodate the data (Figures 1, 5A and 5 B, Gao); create a reference to the data element (column 5 lines 1 – 9, Gao) store the reference in at least one of a header pointer and a forward pointer included in a preceding data element and store the data in the data element (column 1 lines 29 – 43, Gao); data service module that, when executed by a processor, minimally causes the processor to: recognize a data request from a first output unit to the exclusion of all other data requests (column 3 lines 51 – 62, Gao); provide data to a first output unit from a data element according to a data element reference and also advance the data element reference to a subsequent data element while a breakpoint is not encountered (column 2 lines 31 – 38, Gao); mark a subsequent data element as in-use when a breakpoint is encountered (column 3 lines 39 – 50, Gao); create a recommencement reference to a subsequent data element column 4 lines 36 – 49 and 62 – 67, Gao discloses the breakpoint being encountered using figure 5 A&B using elements 510 and 525, creating a recommencement reference to a subsequent element in Gao is clearly

found in column 2 lines 46 – 58, column3 lines 9 – 16 and lines 56 – 59. when the breakpoint has been marked and the flag set as in-use the recommencement point is the one at breakpoint when the flag is unset or search for another container is made according the algorithm in figure 5A&B); and enable recognition of other data requests (column 4 lines 60 – 66, Gao).

Gao however does not disclose the relinquishing of control as a requirement as explicitly being claimed in definition of breakpoint.

Marcotte discloses the condition where relinquishing of control may be required and this is explained in figure 2 and column 7 lines 26 – 65. Here Marcotte discloses how the lock that is help can be in a priority state and thus the queue would have to wait, similarly if the queue that was requiring the lock was in priority state the current item would have to relinquish the lock.

It would have been obvious to one of ordinary skill in the art of data processing at the time of the present invention to combine the teachings of cited references because both inventions are directed to the same field of study of lists and locks in data processing. Furthermore, the state of priority and the lock and its release capabilities of Marcotte reduce lock contention of Gao's method (column 4 lines 4 – 39, Marcotte).

6. Claims 14 – 18 are rejected under the same rationale as claim 13 above. For further citations and references see below.

With respect to claim 14,

Gao as modified discloses the computer readable medium of claim 13 wherein the data service module, when executed by a processor, further minimally causes the processor to: recognize a data request from a first output unit to the exclusion of all other data requests (column 2 lines 46 – 54, Gao); and provide data to a first output unit from a data element according to the commencement reference (column 5 lines 1 – 9, Gao).

With respect to claim 15,

Gao as modified discloses the computer readable medium of claim 13 wherein the data service module causes a processor to create a commencement reference by minimally causing the processor to: retrieve a pointer to a data element subsequent to a current data element (column 2 lines 64 – 67 and column 3 lines 1 – 8, Gao); determine an identifier associated with a data request received from a first output unit and store the retrieved pointer and the determined identifier in an associative manner (column 3 lines 64 – 67 and column 4 lines 1 – 9, Gao).

With respect to claim 16,

Gao as modified discloses the computer readable medium of claim 13 wherein the data service module causes a processor to mark a subsequent data element as in-use by minimally causing the processor to increment a use counter included in a subsequent data element (figure 3 and column 3 lines 39 – 51, Gao).

With respect to claim 17,

Gao as modified discloses the computer readable medium of claim 13 wherein the data service module further minimally causes the processor to receive a delete data request from an output unit by minimally causing the processor to: determine if a data element to be deleted is in-use (column 5 lines 10 – 21, Gao); update a commencement reference to refer to a data element that is subsequent to the data element to be deleted (Table 14, Gao); and delete the data element according to the received delete data request (column 5 lines 25 – 43, Table 14, Gao).

With respect to claim 18,

Gao as modified discloses the computer readable medium of claim 17 wherein the data service module causes the processor to update a commencement reference by minimally causing the processor to: discover a pointer according to a data request identifier (column 5 lines 1 – 9, Gao); and replace the pointer with a pointer to a data element that is subsequent to the data element to be deleted (column 5 lines 10 – 17, Gao).

With respect to claim 19,

Gao discloses an apparatus for storing and retrieving data comprising: means for locking a linked list (column 2 lines 46 – 54, Gao); means for retrieving data from an element in the linked list and also advancing to a subsequent element while a

breakpoint is not encountered (Figure 3, column 3 lines 51 – 59, Gao); means for marking the subsequent element in the linked-list as in-use when a breakpoint is encountered (column 3 lines 39 – 50, Gao); means for creating a recommencement reference to the subsequent element (column 4 lines 36 – 49 and 62 – 67, Gao discloses the breakpoint being encountered using figure 5 A&B using elements 510 and 525, creating a recommencement reference to a subsequent element in Gao is clearly found in column 2 lines 46 – 58, column3 lines 9 – 16 and lines 56 – 59. when the breakpoint has been marked and the flag set as in-use the recommencement point is the one at breakpoint when the flag is unset or search for another container is made according the algorithm in figure 5A&B)); and means for unlocking the linked list (column 4 lines 60 – 66, Gao).

Gao however does not disclose the relinquishing of control as a requirement as explicitly being claimed in definition of breakpoint.

Marcotte discloses the condition where relinquishing of control may be required and this is explained in figure 2 and column 7 lines 26 – 65. Here Marcotte discloses how the lock that is help can be in a priority state and thus the queue would have to wait, similarly if the queue that was requiring the lock was in priority state the current item would have to relinquish the lock.

It would have been obvious to one of ordinary skill in the art of data processing at the time of the present invention to combine the teachings of cited references because both inventions are directed to the same field of study of lists and locks in data

processing. Furthermore, the state of priority and the lock and its release capabilities of Marcotte reduce lock contention of Gao's method (column 4 lines 4 – 39, Marcotte).

7. Claims 20 – 22 are rejected under the same rationale as claim 19 above. For further citations and references see below.

With respect to claim 20,

Gao as modified discloses the apparatus of claim 19 further comprising: means for locking the linked list (column 2 lines 46 – 54, Gao); means for determining a subsequent element in the linked list according to the recommencement reference (column 5 lines 1 – 9, Gao); and means for retrieving data from the determined subsequent element (column 5 lines 10 – 17, Gao).

With respect to claim 21,

Gao as modified discloses the apparatus of claim 19 further comprising a means for deleting an element in the linked-list (column 5 lines 10 – 21, Gao).

With respect to claim 22,

Gao as modified discloses the apparatus of claim 21 wherein the means for deleting an element comprises: means for determining if the element to be deleted is in-use (column 5 lines 10 – 21, Gao); means for updating a reference to the element to refer to a subsequent element in the linked list when the element in is in-use (Table 14, Gao); and means for deleting the element (column 5 lines 25 – 43, Table 14, Gao).

(10) Response to Argument

Appellant argues regarding claims 1, 7, 13, 19, that the combination does not teach "marking the subsequent element in the linked-list as in-use when a breakpoint is encountered" and "creating a commencement reference to the subsequent element".

In response to the Appellant's argument, the examiner contends that the in rejection with references of record, Gao in combination of Marcotte disclose and teach the "marking the subsequent element in the linked-list as in use after encountering a breakpoint". This teaching in Gao is found in figure 3 and column 3 lines 39 – 67. Furthermore, the instant application defines breakpoint as when control is relinquished and the citations are in parallel with that understanding.

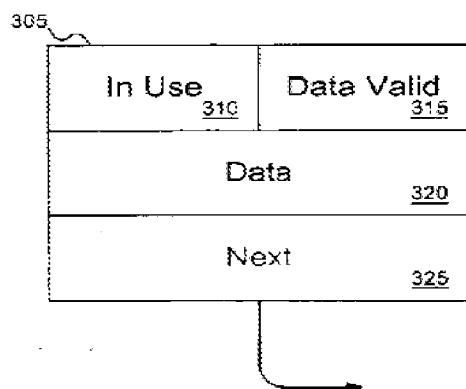
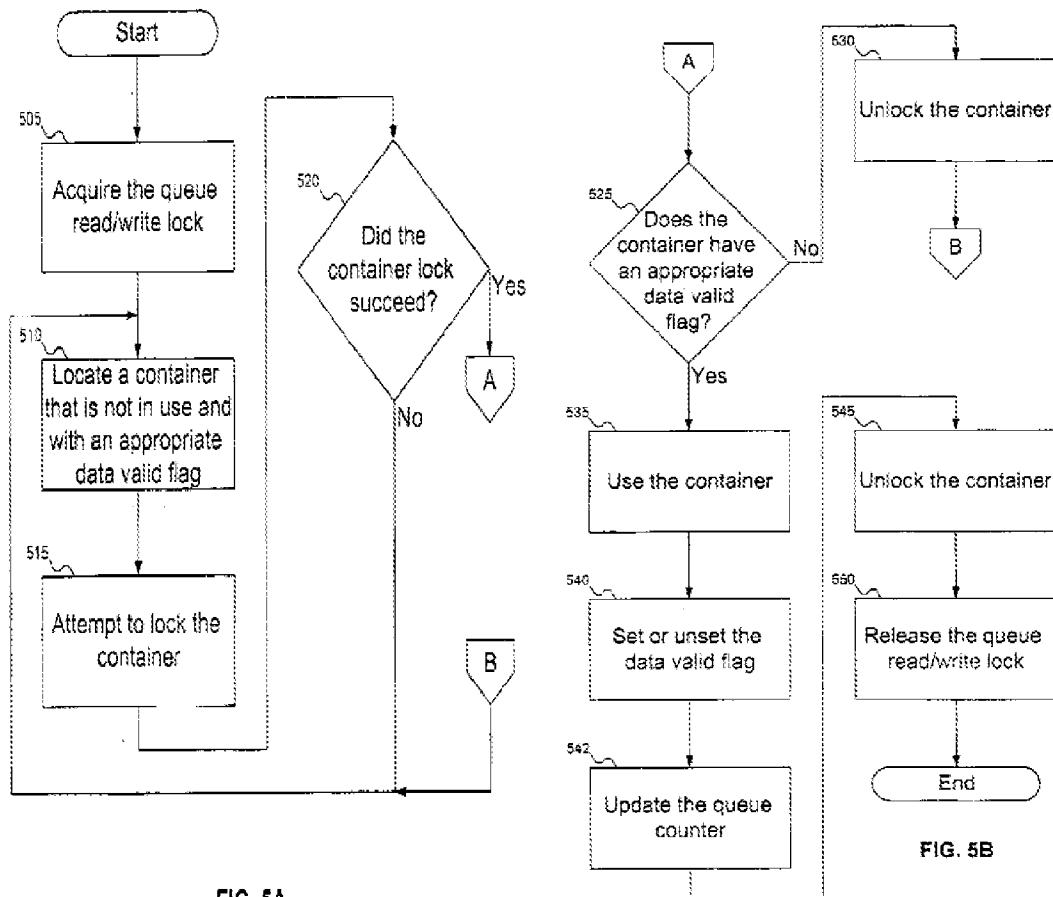


FIG. 3

Furthermore, in column 4 lines 36 – 40 and 62 – 67, Gao discloses the breakpoint being encountered using figure 5 A&B using elements 510 and 525.



Clearly from the figure and the cited text a breakpoint is encountered when a container that is attempted to being used and is locked and set to be in the in-use state and thus is marked. Furthermore, the examiner contends that Gao discloses “creating a commencement reference to a subsequent element”. This teaching in Gao is clearly found in column 2 lines 46 – 58, column3 lines 9 – 16 and lines 56 – 59. when the breakpoint has been marked and the flag set as in-use the commencement point is the one at breakpoint when the flag is unset or search for another container is made

according the algorithm in figure 5A&B cited above. The recommencement is clearly explained and showed by the pointer in the reference.

Appellant argues regarding claim 5, that the combination does not teach "updating a recommencement reference to the element to refer to a data element that is subsequent to data element to be deleted when the element is in-use".

In response to the Appellant's argument, the examiner contends that Gao in combination with Marcotte discloses and teaches the " updating a recommencement reference to the element to refer to a data element that is subsequent to data element to be deleted when the element is in-use ". This teaching in Gao is found in column 5 lines 10 – 21, Table 14, Gao. Gao however does not disclose the relinquishing of control as a requirement as explicitly being claimed in definition of breakpoint. Marcotte discloses the condition where relinquishing of control may be required and this is explained in figure 2 and column 7 lines 26 – 65. Here Marcotte discloses how the lock that is held can be in a priority state and thus the queue would have to wait, similarly if the queue that was requiring the lock was in priority state the current item would have to relinquish the lock.

(11) Related Proceeding(s) Appendix

No decision rendered by a court or the Board is identified by the examiner in the Related Appeals and Interferences section of this examiner's answer.

For the above reasons, it is believed that the rejections should be sustained.

Respectfully submitted,

/Navneet K. Ahluwalia/

Examiner, Art Unit 2166

Conferees:

/Hosain T Alam/

Supervisory Patent Examiner, Art Unit 2166

/Mohammad Ali/

Supervisory Patent Examiner, Art Unit 2158